EXP 2

CONVERSION FROM REGULAR EXPRESSION TO NFA

Sai Ram Pendyala

RA1911003010696

AIM:

To write a program for converting Regular Expression to NFA.

ALGORITHM:

1. Start
2. Get the input from the user
3. Initialize separate variables and functions for Postfix , Display and NFA
4. Create separate methods for different operators like +,*, .
5. By using Switch case Initialize different cases for the input
6. For ' . ' operator Initialize a separate method by using various stack functions do the same for the other operators like ' * ' and ' + '.
7. Regular expression is in the form like a.b (or) a+b
8. Display the output
9. Stop

PROGRAM:

```
transition_table = [ [0]*3 for _ in range(20) ]
re = input("Enter the regular expression : ")
re += " "

i = 0
j = 1
while(i<len(re)):
    if re[i] == 'a':
        try:
            if re[i+1] != '|' and re[i+1] !='*':
                transition_table[j][0] = j+1
                j += 1
            elif re[i+1] == '|' and re[i+2] =='b':
                transition_table[j][2]=((j+1)*10)+(j+3)
                j+=1
                transition_table[j][0]=j+1
                j+=1
                transition_table[j][2]=j+3
                j+=1
                transition_table[j][1]=j+1
                j+=1
                transition_table[j][2]=j+1
                j+=1
                i=i+2
            elif re[i+1]=='*':
```

```
                transition_table[j][2]=((j+1)*10)+(j+3)
                j+=1
                transition_table[j][0]=j+1
                j+=1
                transition_table[j][2]=((j+1)*10)+(j-1)
                j+=1
        except:
            transition_table[j][0] = j+1

    elif re[i] == 'b':
        try:
            if re[i+1] != '|' and re[i+1] !='*':
                transition_table[j][1] = j+1
                j += 1
            elif re[i+1]=='|' and re[i+2]=='a':
                transition_table[j][2]=((j+1)*10)+(j+3)
                j+=1
                transition_table[j][1]=j+1
                j+=1
                transition_table[j][2]=j+3
                j+=1
                transition_table[j][0]=j+1
                j+=1
                transition_table[j][2]=j+1
                j+=1
                i=i+2
            elif re[i+1]=='*':
                transition_table[j][2]=((j+1)*10)+(j+3)
                j+=1
                transition_table[j][1]=j+1
                j+=1
                transition_table[j][2]=((j+1)*10)+(j-1)
                j+=1
        except:
            transition_table[j][1] = j+1

    elif re[i]=='e' and re[i+1]!='|'and re[i+1]!='*':
        transition_table[j][2]=j+1
        j+=1

    elif re[i]==')' and re[i+1]=='*':

        transition_table[0][2]=((j+1)*10)+1
        transition_table[j][2]=((j+1)*10)+1
        j+=1

    i +=1
```

```
print ("Transition function:")
print("s  a  b  e\n")
for i in range(j):
    if(transition_table[i][0]!=0):
        print("q[{0},a]-->{1}".format(i,transition_table[i][0]))
    if(transition_table[i][1]!=0):
        print("q[{0},b]-->{1}".format(i,transition_table[i][1]))
    if(transition_table[i][2]!=0):
        if(transition_table[i][2]<10):
            print("q[{0},e]-->{1}".format(i,transition_table[i][2]))
        else:
            print("q[{0},e]-->{1} &
{2}".format(i,int(transition_table[i][2]/10),transition_table[i][2]%10))
```

RESULT: